

Datenbanken und Datensicherheit

Prof. Dr. rer. nat. Peter Gerwinski

2. November 2023

Datenbanken und Datensicherheit

<https://gitlab.cvh-server.de/pgerwinski/dbs>

- 1 Einführung**
- 2 Kurzeinführung Unix**
- 3 Kurzeinführung TCP/IP**
 - 3.1 IP-Adressen
 - 3.2 MAC-Adressen
 - 3.3 TCP- und UDP-Ports
 - 3.4 TCP-Protokolle
 - 3.5 Routing
 - 3.6 Netzwerkanalyse
 - 3.7 SSH
 - 3.8 X11
- 4 Relationale Datenbanken**

...



Änderungen
vorbehalten

3.1 IP-Adressen

- `ip addr` (Linux)
 `ifconfig` (Unix allgemein)
 `ipconfig` (MS Windows)
- `ip addr add <Netz>`
- `ip link`
- `ping <IP-Adresse>`

```
# ifconfig
```

```
lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
    inet 127.0.0.1  netmask 255.0.0.0
    inet6 ::1  prefixlen 128  scopeid 0x10<host>
    [...]
```

```
wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.42.101  netmask 255.255.255.0
                                broadcast 192.168.42.255
    ether be:3f:ca:aa:7e:51 txqueuelen 1000  (Ethernet)
    [...]
```

3.1 IP-Adressen

IPv4-Adressen:

- 32 Bit
- dezimal, 4 Gruppen zu je 8 Bit (0–255), durch Punkte getrennt

IPv6-Adressen:

- 128 Bit
- hexadezimal, 8 Gruppen zu je 4 Hex-Ziffern, durch Doppelpunkte getrennt
- Führende Nullen dürfen weggelassen werden.
- Zwei Doppelpunkte bedeuten: Mit Nullen auffüllen.
- Literatur und Beispiel: <https://de.wikipedia.org/wiki/IPv6>

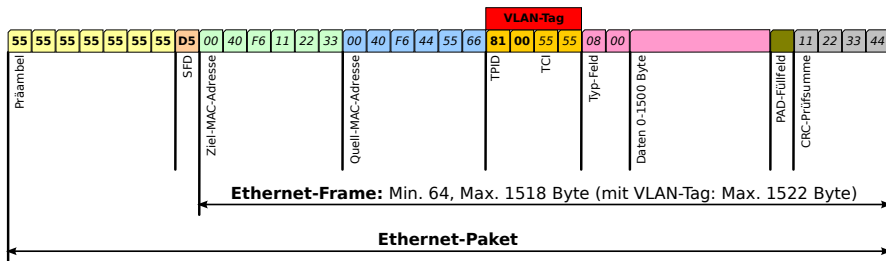
3.2 MAC-Adressen

MAC = Media Access Control

MAC-Adresse = Hardware-Adresse = Ethernet-Adresse

- ip neig
arp

Ethernet-Frame:

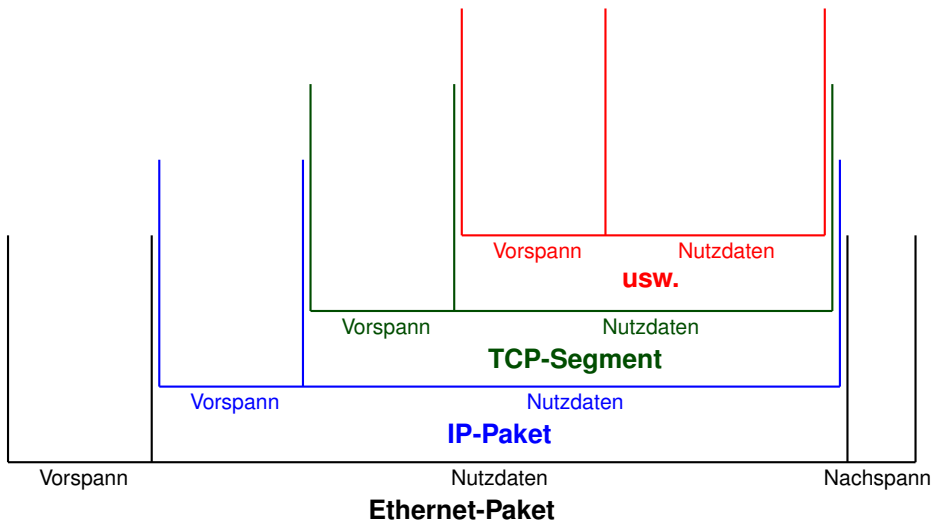


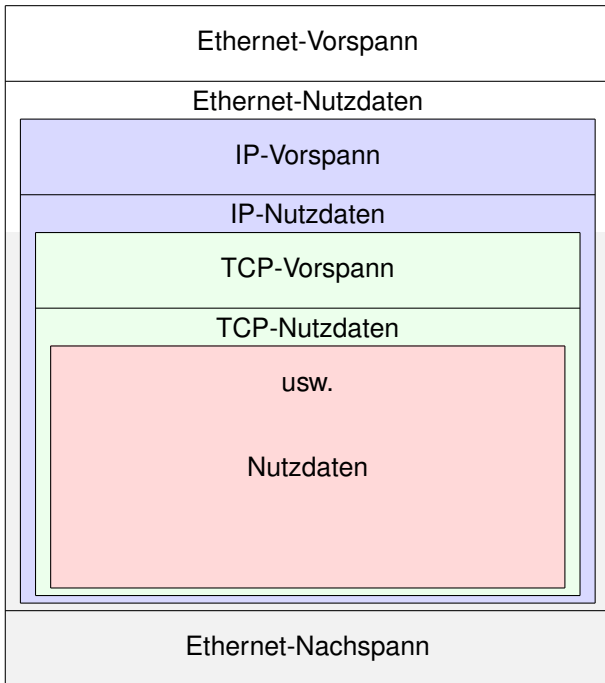
3.3 TCP- und UDP-Ports

- `nc <IP> <Port>`
Verbindung zu Programm $\langle \text{Port} \rangle$ auf Rechner $\langle \text{IP} \rangle$ aufnehmen
- `nc -l <Port>` oder `nc -p <Port> -l`
auf eingehende Verbindungen warten („lauschen“)
- TCP-Ports: Verbindungskonzept, Netzwerk prüft
- UDP-Ports: einzelne Pakete, Anwendung muß selbst prüfen
- ICMP: keine Ports, nur Rechner:
Erreichbarkeit, Eigenschaften der Übertragung

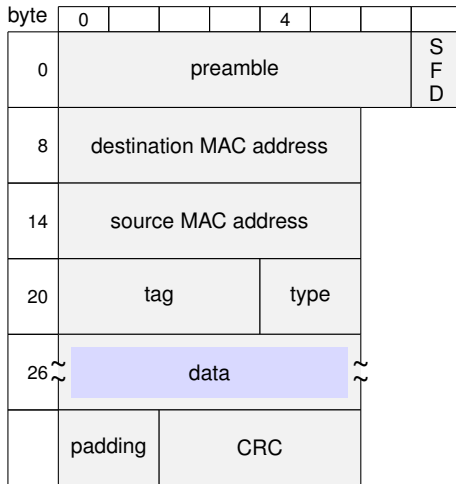
Anwendung: HTTP, SMTP, ...
Transport: TCP-/UDP-Ports, ICMP
Internet: IP-Adresse
Netzwerkzugang: Hardware-/MAC-Adresse

Protokollstapel:

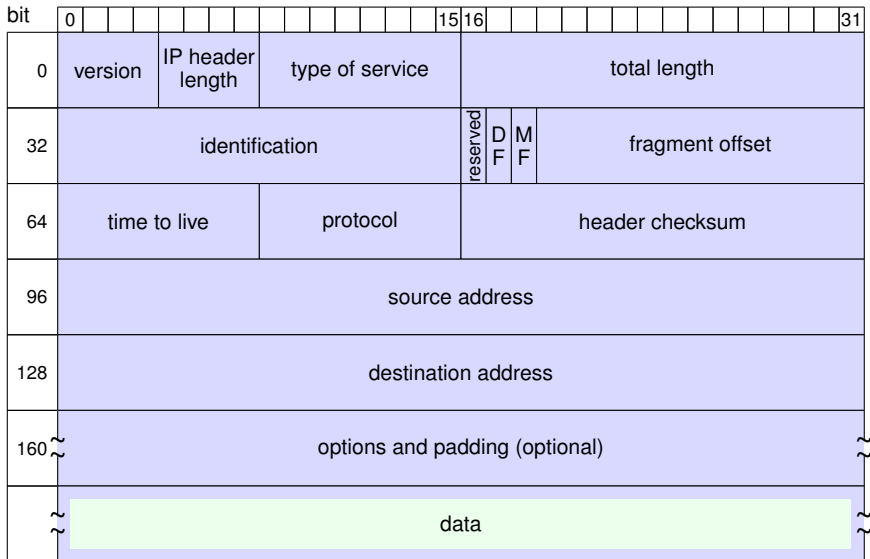




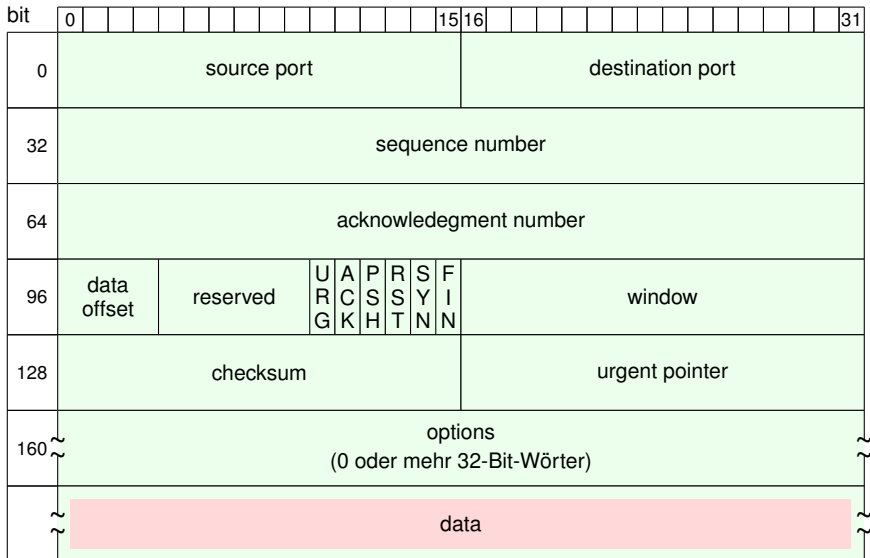
Ethernet-Frame: transportiert IP-Paket



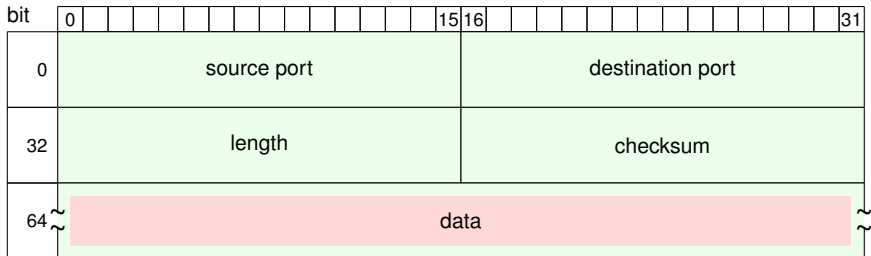
IP-Paket: transportiert TCP-Datagramm, UDP- oder ICMP-Paket



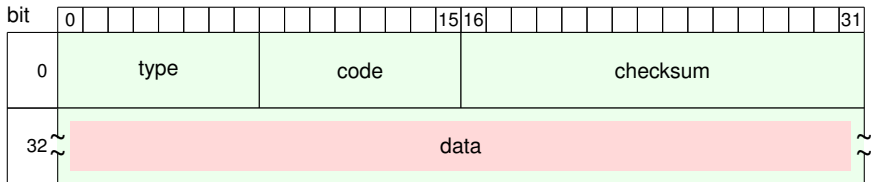
TCP-Datagramm: transportiert Anwendungsdaten



UDP-Paket: transportiert Anwendungsdaten



ICMP-Paket: transportiert Steuernachrichten



3.4 TCP-Protokolle

- **SMTP**

```
HELO cassini
```

```
MAIL FROM: <example@example.com>
```

```
RCPT TO: <beispiel@example.de>
```

(E-Mail-Header – Teil der Nutzdaten)

```
From: Eddie Example <example@example.com>
```

```
To: Bert Beispiel <beispiel@example.de>
```

```
Subject: Hello, world!
```

(Leerzeile)

```
Hi, there!
```

```
.
```

3.4 TCP-Protokolle

- **SMTP**

HELO cassini

MAIL FROM: <example@example.com>

RCPT TO: <beispiel@example.de>

(E-Mail-Header – Teil der Nutzdaten)

From: Eddie Example <example@example.com>

To: Bert Beispiel <beispiel@example.de>

Subject: Hello, world!

(Leerzeile)

Hi, there!

.

- **HTTP**

GET / HTTP/1.1

Host: www.hs-bochum.de

(Leerzeile)

URL: Schema://Benutzer:Passwort@Rechner:port/Pfad?Query#Fragment

3.4 TCP-Protokolle

- **SMTP**

HELO cassini

MAIL FROM: <example@example.com>

RCPT TO: <beispiel@example.de>

(E-Mail-Header – Teil der Nutzdaten)

From: Eddie Example <example@example.com>

To: Bert Beispiel <beispiel@example.de>

Subject: Hello, world!

(Leerzeile)

Hi, there!

.

- **HTTP**

GET / HTTP/1.1

Host: www.hs-bochum.de

(Leerzeile)

URL: Schema://Benutzer:Passwort@Rechner:port/Pfad?Query#Fragment

- Protokolle „mal eben“ selbst schreiben: `nc` `-c` oder `inetd`

3.5 Routing

- `ip route` (Linux)
`route` (MS-Windows, Unix)
`netstat -nr` (MacOS)

```
# route -n
```

Kernel-IP-Routentabelle

Ziel	Router	Genmask	[...]	Iface
0.0.0.0	192.168.42.1	0.0.0.0	[...]	wlan0
169.254.0.0	0.0.0.0	255.255.0.0	[...]	wlan0
192.168.42.0	0.0.0.0	255.255.255.0	[...]	wlan0

Netzmaske:

Wenn nach Und-Verknüpfung mit IP-Adresse gleich, —→ im gleichen Netz

255.255.240.0 ist dasselbe wie /20

(20 Bit sind 1; die restlichen 12 Bit sind 0)

3.6 Netzwerkanalyse

- `tcpdump`
- `wireshark`
- `ettercap`

3.7 SSH

- SSH <Rechner>
- -C: Komprimierung
- -L: lokalen Port auf Remote-Port umleiten
- -R: Remote-Port auf lokalen Port umleiten

3.8 X11

- Grafik-Bildschirm und Eingabegeräte über's Netz
- `DISPLAY`-Variable: X-Server: Rechner und Bildschirm
- `ssh -X`: X11-Forwarding

Datenbanken und Datensicherheit

<https://gitlab.cvh-server.de/pgerwinski/dbs>

1 Einführung

2 Kurzeinführung Unix

3 Kurzeinführung TCP/IP

3.1 IP-Adressen

3.2 MAC-Adressen

3.3 TCP- und UDP-Ports

3.4 TCP-Protokolle

3.5 Routing

3.6 Netzwerkanalyse

3.7 SSH

3.8 X11

4 Relationale Datenbanken

...



Änderungen
vorbehalten